Optimal knot selection for least-squares fitting of noisy data with spline functions

Jerome Blair
National Security Technologies, LLC
P. O. Box 98521, M/S NLV-071
Las Vegas, NV 89193-8521
j.blair@ieee.org

Abstract *– An automatic data-smoothing algorithm for data from digital oscilloscopes is described. The algorithm adjusts the bandwidth of the filtering as a function of time to provide minimum mean squared error at each time. It produces an estimate of the root-mean-square error as a function of time and does so without any statistical assumptions about the unknown signal. The algorithm is based on least-squares fitting to the data of cubic spline functions.*

Keywords *– data smoothing, multiresolution analysis, spline functions, estimation*

I. INTRODUCTION

This is the second in a series of papers on a particular class of practical methods for extracting an accurate estimate of a signal from noisy measurements. The problem, in the simplest form that will be considered, is that a signal $s(t)$ is measured at uniformly spaced discrete times, τ_i, for $i = 1$ to N. The measurements have random noise with known statistics. Throughout this paper it will be assumed that the measurement noise is white. However, the approach described here has been successfully used for problems in which the noise is not white nor even stationary, and the sampling very non-uniform. This problem was first systematically studied in its modern form in [1]-[3], though closely related problems were studied by Gauss [4] as far back as 1804. The measured signal is represented as $\boldsymbol{y} = \boldsymbol{s} + \boldsymbol{e}$, where $\boldsymbol{s}$ is the true signal and $\boldsymbol{e}$ is the vector of measurement errors. We estimate the signal with $\hat{\boldsymbol{s}}$, where

$$\hat{\boldsymbol{s}} = \boldsymbol{P}_{\boldsymbol{K}}\boldsymbol{y} = \boldsymbol{P}_{\boldsymbol{K}}\boldsymbol{s} + \boldsymbol{P}_{\boldsymbol{K}}\boldsymbol{e}\,, \tag{1}$$

where $\boldsymbol{P}_{\boldsymbol{K}}$ is a linear operator that is applied to the data to give an accurate estimate of the signal. The operator, $\boldsymbol{P}_{\boldsymbol{K}}$, is designed to smooth, or filter, the data to reduce the noise while not distorting the signal too much. $\boldsymbol{P}_{\boldsymbol{K}}$ is the operator that maps the data onto the cubic spline function with knots specified in the vector, $\boldsymbol{K}$, that gives the least-squares fit to the data. The error in the recovered signal is given by $\boldsymbol{e}_s$ with

$$\boldsymbol{e}_s = \boldsymbol{s} - \hat{\boldsymbol{s}} = \boldsymbol{s} - \boldsymbol{P}_{\boldsymbol{K}}\boldsymbol{s} - \boldsymbol{P}_{\boldsymbol{K}}\boldsymbol{e} = (\boldsymbol{I} - \boldsymbol{P}_{\boldsymbol{K}})\boldsymbol{s} - \boldsymbol{P}_{\boldsymbol{K}}\boldsymbol{e}\,, \tag{2}$$

where $\boldsymbol{I}$ is the identity operator. In Equation (2), there are two sources of error: one resulting from the term $(\boldsymbol{I}-\boldsymbol{P}_{\boldsymbol{K}})\boldsymbol{s}$ and one resulting from the term $\boldsymbol{P}_{\boldsymbol{K}}\boldsymbol{e}$. In this paper, the first term is called the F-error (which could equally mean fitting error or filtering error). It is the error the smoothing operation introduces in the absence of measurement errors. The second term is called the R-error, which is the error in the reconstructed signal caused by the measurement errors. The first paper in this series, [5], gave a method for estimating the standard deviation, $\sigma_F(t)$, of the F-error, $e_F(t)$. The estimation of the standard deviation, $\sigma_R(t)$, of the R-error, $e_R(t)$, is a standard statistical calculation, which was also given in [5].

In this paper the two error estimates are combined to give an estimate of the standard deviation of the total error

$$\sigma_T(t) = \left(\sigma_F^2(t) + \sigma_R^2(t)\right)^{1/2}, \tag{3}$$

and an algorithm is given to iteratively improve the knot sequence, $\boldsymbol{K}$, to minimize this error at each of a sufficiently large set of values of t. This gives a minimum mean-squared error (MMSE) estimate for the signal. In the previous literature ([1]-[3],[6]) a MMSE estimate requires an a priori statistical distribution for the signal. The method presented here gives both the MMSE estimate and an estimate of the root-mean-square (rms) error as a function of time—without any prior distribution for the signal. The approaches using smoothing splines, which are discussed in [7], yield an "optimal" signal estimate without a prior distribution but do not produce error estimates nor use variable smoothing as a function of time, as does the method here.

Implicit in some of the calculations is that the sampling rate is a factor of five more than the minimum necessary. This is because facts about spline functions that were developed for continuous time functions are being applied to discrete time functions. This assumption is relevant to practical problems, because in recent years the sampling rate and bandwidth of digital oscilloscopes has been increasing rapidly while the noise level has remained constant or deteriorated. This makes the situation of high sampling rate and high noise level one of importance. It is also assumed that the unknown signal has four derivatives, but no assumptions are made about the magnitudes of the derivatives.

II. BACKGROUND AND NOTATION

The smoothing operators used are based on cubic spline functions. Let the interval over which the signal is measured be $T_1 \le t \le T_2$, and let $\boldsymbol{K}$ be a sequence of time values, t_k, for $k = 1$ to $n < N$ satisfying $t_1 = T_1$, $t_{k+1} > t_k$ and $t_n = T_2$. A *cubic spline* with knots, $\boldsymbol{K}$, is a function defined on the interval $[T_1, T_2]$ that is a polynomial of degree three on each sub-interval of the form $[t_k, t_{k+1}]$ and has two continuous derivatives throughout the interval $[T_1, T_2]$. The symbol, S_K, denotes the vector space of cubic spline functions with knot sequence K. Because of the continuity requirement on the second derivatives, the dimension of S_K is $n + 2$. These functions and many algorithms for dealing with them are described in [8]. The algorithms in [8] are given in FORTRAN. The author used the MATLAB implementation of these algorithms [9].

The estimate, $\hat{\boldsymbol{s}}$, for the signal is the least-squares fit to the data by a cubic spline with a selected knot sequence $\boldsymbol{K}$. Precisely,

$$\hat{s} \in S_K \text{ and minimizes } \sum_{i=1}^{N} (\hat{s}(\tau_i) - y_i)^2 . \tag{4}$$

The solution to this problem depends linearly on the data and is written as

$$\hat{\boldsymbol{s}} = \boldsymbol{P}_{\boldsymbol{K}} \boldsymbol{y} . \tag{5}$$

Of particular importance is how close the knots are to their neighbors. This is measured with the quantities

$$\Delta_k = t_{k+1} - t_k \text{ for } 1 \le k \le n-1 \text{ and } t_k \in K . \tag{6}$$

The quantity Δ_k is called the *mesh size* of the k^{th} interval. The knot sequences are restricted to those for which Δ does not vary too rapidly; specifically it is required that

$$1/2 \le \Delta_{k+1} / \Delta_k \le 2 . \tag{7}$$

The operation (5) is a time-varying low-pass filter with bandwidth of (see [10] and [11])

$$BW \cong \frac{1}{2\Delta_k} \text{ for } t \text{ near the knot } t_k . \tag{8}$$

The optimal filtering will have the mesh size smaller where larger bandwidth is required to represent the signal and larger where smaller bandwidth is required.

III. QUALITATIVE DEPENDENCE OF THE ERRORS ON THE MESH SIZE

The algorithm presented later for determining the optimum distribution of knots depends on qualitative relations between the magnitudes of the two errors at any particular time and the mesh size near that time. The important result is:

$$\begin{aligned} &\text{For } t \in [t_k, t_{k+1}],\ \sigma_R(t) \cong C_R \Delta_k^{-1/2}, \text{ and} \\ &\sigma_F(t) \cong C_F s^{(4)}(t) \Delta_k^4 = C'_F(t) \Delta_k^4, \end{aligned} \tag{9}$$

where $s^{(4)}(t)$ is the unknown fourth derivative of the signal, and C_R and C_F are constants. For uniform mesh size the constants can be evaluated; for non-uniform meshes they depend on the ratios of the mesh size with the nearby mesh sizes. Condition (7) guarantees that the results remain approximately true. The proof of this result in the generality stated above is very long and tedious and beyond the scope of this paper. However, an indication of the source of the results will be given here.

The first part of (9) follows form (8) and the fact that a filter reduces white noise by a factor proportional to the square root of the bandwidth. The second part follows from the fact that fitting with cubic spline functions has fourth order accuracy (see [8]) and that the value of the fitted spline at any time, t, depends (approximately) only on the data near t.

It should be noted that none of the error estimates calculated by the method described in this paper depend on (9). These two equations are only used as a heuristic in the iterative procedure for setting the knot locations.

IV. ERROR RATIO AT OPTIMUM KNOT SPACING

The algorithm for determining the mesh size as a function of time, which will be described in detail later, involves selecting an initial distribution of knots, estimating the standard deviations for the two error sources for the distribution (as a function of time). Then, a new mesh size as a function of time is calculated from the estimated standard deviations. This section describes the derivation of the new mesh. It turns out that (9) implies that the ratio of the two standard deviations is a fixed known quantity when the mesh size is optimal. The derivation will be made for a generalization of (9) that is useful in other situations (e.g., the design of differentiation filters). Let

$$\sigma_1 = C_1 \Delta^{-p_1},\ \sigma_2 = C_2 \Delta^{p_2}, \text{ and } \sigma_T^2 = \sigma_1^2 + \sigma_2^2 , \tag{10}$$

where the Cs and ps are all positive constants.
Differentiating to minimize σ_T gives

$$\begin{aligned} 0 = \frac{d\sigma_T^2}{d\Delta} &= \frac{d}{d\Delta}\left(C_1^2 \Delta^{-2p_1} + C_2^2 \Delta^{2p_2} \right) \\ &= -2p_1 C_1^2 \Delta^{-2p_1 - 1} + 2p_2 C_2^2 \Delta^{2p_2 - 1} = -2p_1 \frac{\sigma_1^2}{\Delta} + 2p_2 \frac{\sigma_2^2}{\Delta}. \end{aligned}$$

This yields

$$\frac{\sigma_2}{\sigma_1} = \sqrt{\frac{p_1}{p_2}} \text{ at optimum } \Delta. \tag{11}$$

This gives the ratio of the two error estimates at the optimum mesh size, and it is independent of the constants in (10) and of the mesh size. The following notation will be useful:

$$\rho(\Delta) = \frac{\sigma_2(\Delta)}{\sigma_1(\Delta)} , \tag{12}$$

$$\rho_\lambda = \lambda \sqrt{\frac{p_1}{p_2}}, \text{ and } \Delta_\lambda \text{ stands for the solution of } \rho(\Delta_\lambda) = \rho_\lambda . \tag{13}$$

From (10)

$$\rho(\Delta) = \frac{C_2}{C_1}\Delta^{p_1+p_2}, \tag{14}$$

which gives

$$\frac{\rho_\lambda}{\rho(\Delta)} = \left(\frac{\Delta_\lambda}{\Delta}\right)^{p_1+p_2}, \text{ or } \Delta_\lambda = \Delta\left(\frac{\rho(\Delta)}{\rho_\lambda}\right)^{-\frac{1}{p_1+p_2}} \tag{15}$$

Thus, if the value of $\rho(\Delta)$ is known for any value of Δ, then (15) can be used to calculate an improved mesh size, Δ_λ. Although Δ_1 is the optimum value of Δ, a larger value of λ will be used in the iterative procedure to improve noise immunity. From (15) and (13)

$$\frac{\Delta_1}{\Delta_\lambda} = \left(\frac{\rho_1}{\rho_\lambda}\right)^{\frac{1}{p_1+p_2}} = \left(\frac{1}{\lambda}\right)^{\frac{1}{p_1+p_2}}, \tag{16}$$

which will be used later.

V. KNOT LOCATIONS FROM KNOT SPACING

For a given knot sequence, an estimate is made for σ_F and σ_R for each knot interval. The ratio, $\rho_k = \sigma_F/\sigma_R$, is then calculated for each interval. Then a new target mesh size, $\hat{\Delta}_k$, is calculated for each interval. This section describes how to determine a sequence of knots so that the mesh size at any time is approximately $\hat{\Delta}_k$. This procedure is similar to that appearing in Chapter XII of [8] (page 158). Let

$$f(t) = \frac{1}{\hat{\Delta}_k} \text{ for } t_k \le t < t_{k+1}. \tag{17}$$

The function, $f(t)$, specifies the desired knot density as a function of time. Let

$$F(t) = \int_{T_1}^{t} f(t')dt'. \tag{18}$$

The function, $F(t)$, gives the desired number of knots in the interval $(T_1,t]$. A knot sequence matching this can only be found if $F(T_2)$ is an integer. Let $G(t)$ be the multiple of $F(t)$ that satisfies $G(T_2)$ = round($F(T_2)$), where round() gives the closest integer to its argument. The new knot sequence is then defined by

$$\hat{t}_1 = T_1, \text{ and } \hat{t}_k \text{ is the solution of } G(\hat{t}_k) = k - 1. \tag{19}$$

This places the last knot at T_2.

VI. THE ALGORITHM

This section gives the details of the algorithm in its current stage of development. The areas where more work could be done are the starting and ending of the algorithm. The steps are:

1. Generate an initial knot distribution.
2. Calculate the estimate of σ_F and σ_R for each knot interval using the methods given in [5].
3. For each knot interval, calculate $\hat{\Delta}_k$ using (15) with $\lambda = 5$.
4. Calculate a new knot sequence using (19).
5. Repeat steps 2 through 4 many times.
6. Calculate a new mesh size in each interval based on $\lambda = 1$ using (16).
7. Calculate new knots based on the results of step 6 using (19).
8. Fit the data using these knots and calculate the error estimates.

Comments on some of the individual steps are in the following subsections.

A. Step 1

For the examples in this paper, the initial knot sequence is uniform. In some other applications a person chooses the initial knot sequence through a user interface. The use of a uniform initial knot sequence is problematic. If the initial mesh size is too large, some localized high-frequency features of the signal could be missed. If the initial mesh size is too small, the computation time could be excessive, and noise in the data could cause a failure to converge. We plan a future investigation of the use of a wavelet decomposition of the data for selecting an initial knot distribution.

B. Step 2

The estimate of σ_F is exactly as in [5] and is determined by comparing the fit using two different knot sequences. The estimate of σ_R was determined using an algorithm that is mathematically equivalent to, but much more computationally efficient than, that given in [5]. This algorithm will be described in a later paper. Note that in any case, the estimate of σ_R depends only on the noise standard deviation and the knot sequence. It is independent of the data.

C. Step 3

The use of $\lambda = 5$ means that the target for the ratio of σ_F to σ_R (i.e., the ratio that the iterative procedure attempts to achieve) is five times as large as the optimum value. The optimum value is $1/\sqrt{8} = 0.35$ (from (11)). The analysis in Section VII of [5] shows that the value calculated for σ_F in the presence of noise will have a random error due to the noise, and that the error will have a standard deviation of approximately $0.25\sigma_R$. This means that the calculated ratio, σ_F/σ_R, will have a standard deviation of 0.25, independent of the value of either the numerator or the denominator. If $\lambda = 1$ were used in the iterations, the standard deviation of the calculated ratio would be 71% of its value, giving very poor results. The observed effect of using too low of a value for λ (e.g., $\lambda = 3$) is that the algorithm converges to a knot sequence with randomly placed intervals within which the values of the mesh size are much smaller than optimal.

D. Step 5

No data dependent stopping condition has yet been developed. The examples used in this paper (and other applications) currently iterate for 10 minutes.

E. Step 6

The new target mesh size at each time, t, is obtained (from (16)) by multiplying the mesh size from the last iteration by $(1/5)^{\frac{2}{9}} = 0.70$. This decreases σ_F by a factor of approximately 4 and increases σ_R by approximately 20%.

VII. EXAMPLES

This section contains simulations of two examples. Both examples use the signal

$$s(t) = \begin{cases} 0 \text{ for } t \le 0 \\ erfc\left(\sqrt{\frac{\tau}{t}}\right) \text{ for } t > 0 \end{cases}, \qquad (20)$$

which represents the step response of a skin-effect limited coaxial cable. The value used for τ is 0.2 ns, which corresponds to about 30 m of RG-58 cable. The simulated sampling rate is 40 GSa/s, a typical sampling rate for a modern high-speed digital oscilloscope. The signal was sampled for 1 μs, or 40,000 samples. Figure 1 shows the signal (1 V amplitude) for example 1 along with samples generated with 10 mV rms of noise. This signal was selected for the examples, because the bandwidth required to reproduce it varies significantly with time. The algorithm was carried out starting with an initial uniform knot sequence with mesh size equal to 1 ns.

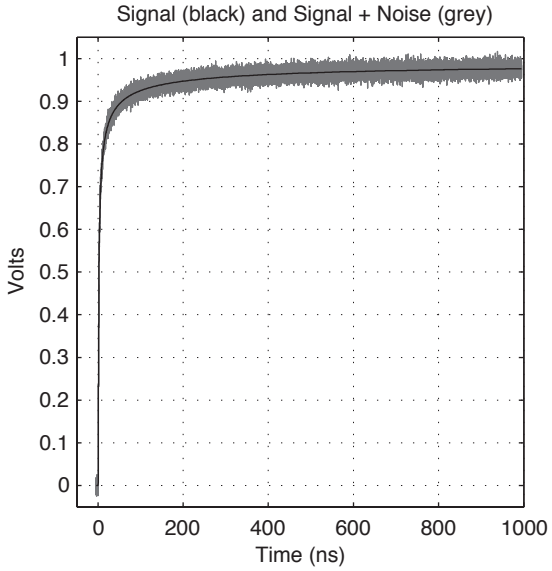
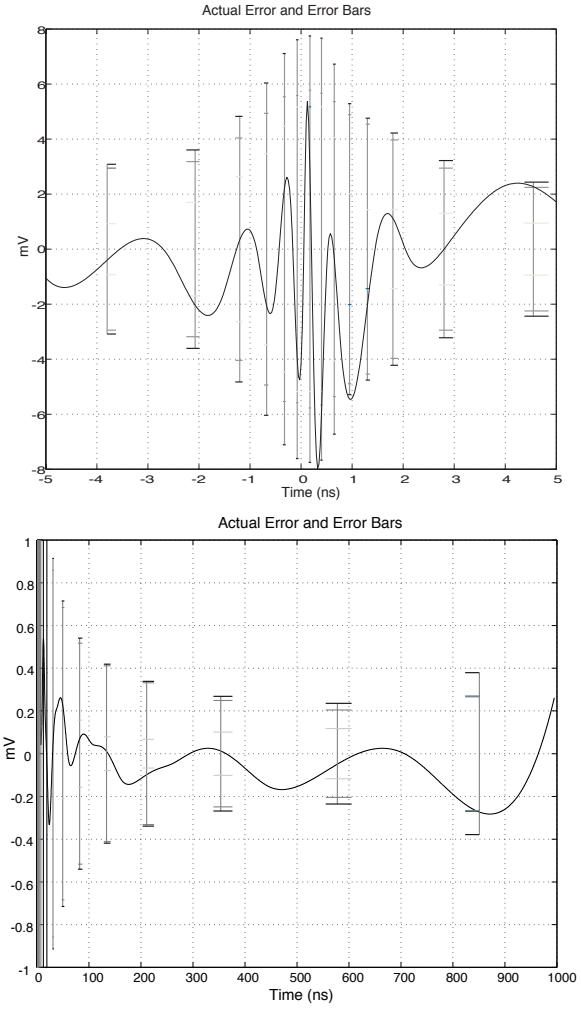


Fig. 1. The signal for example 1 with the simulated noise.

In the examples the term "error" means the difference between the signal and the spline function that results from applying the algorithm described in the previous sections to the noisy data, i.e., the function $\boldsymbol{e_s}$ of equation (2). In an actual measurement situation the error remains unknown; only the error estimates are available. Figures 2 and 3 show the error as a function of time along with the error estimates (two sigma) determined in step 8 of the algorithm. An error bar is shown at the center of each knot interval. On each error bar are three pairs of horizontal lines. The innermost (light grey) is at $\pm 2\sigma_F$. The next pair from the center (grey) is at $\pm 2\sigma_R$, and the outermost pair (black) is at $\pm 2\sigma_T$, as given by (3). Figure 2 shows the results near $t = 0$, while Figure 3 shows them at large t. The actual error is expected to exceed the two-sigma error bars over 5% of the measurement interval.



Figs. 2 and 3. Error as a function of time for example 1 along with computed two-sigma error bars.

The signal for the second example is shown in Figure 4. It is the same signal as in example 1 except that a small glitch, about 5 ns in duration, has been added at t = 200 ns. The glitch is about half the size of the noise. Figure 5 shows the signal along with the spline fit and the error bars—with a

close up of the interval containing the glitch. The algorithm effectively retains the glitch with accurate error bars while filtering the surrounding data. Figures 6 and 7 show the actual error along with the two-sigma error bars, with Figure 7 concentrating on the area near the glitch.

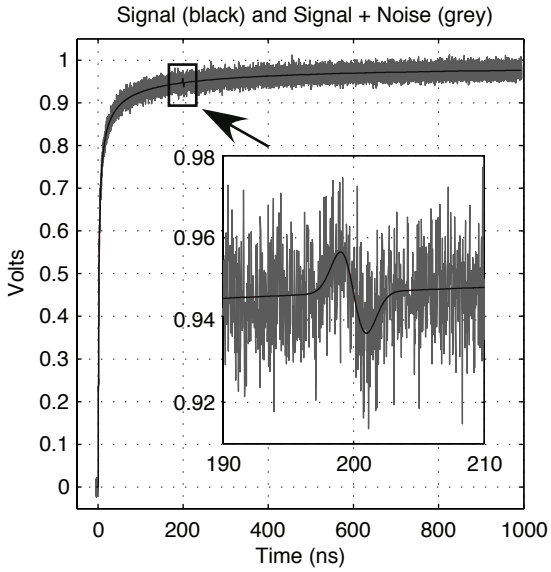


Fig. 4. The signal of example 2 along with the simulated noise.

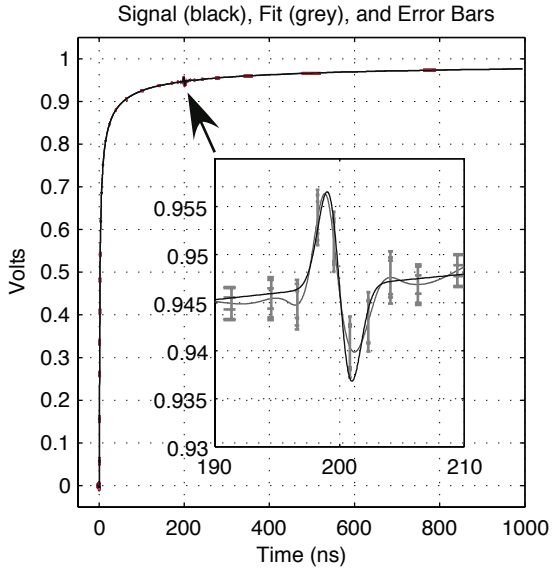
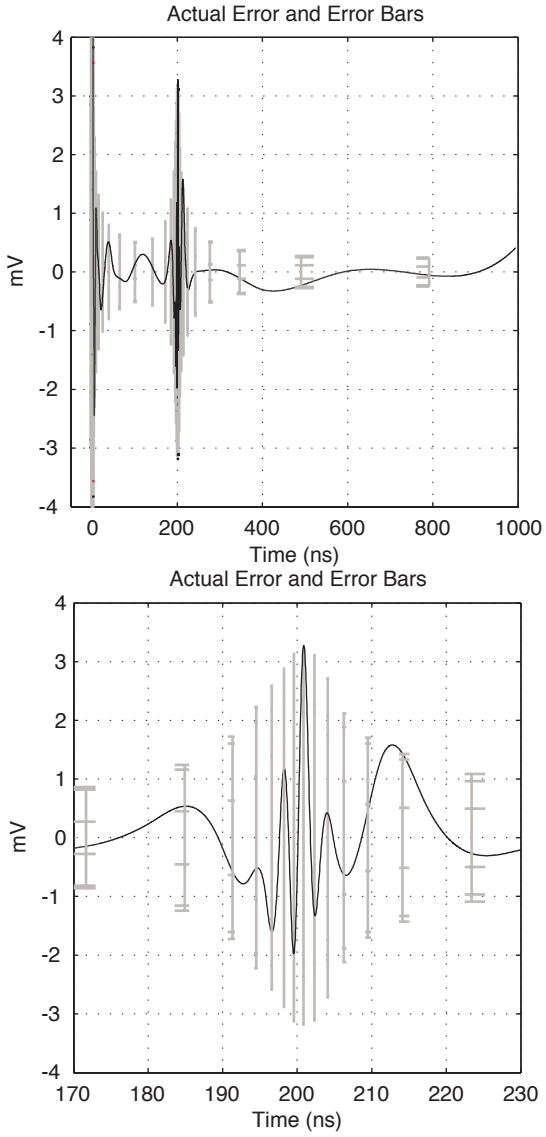


Fig. 5. The signal of example 2 along with the fitted spline function and the error bars.



Figs. 6 and 7. The error as a function of time along with error bars for example 2.

Figure 8 shows the bandwidth of the filter in GHz as a function of time. The bandwidth is calculated from the mesh size using (8). Both examples are shown. Obviously the curve with the peak at 200 ns is for example 2. The bandwidth varies from 2.5 GHz to about 2 Mhz. Note that the presence of the glitch causes the bandwidth at 200 ns to change from 5.5 MHz for example 1 to 450 MHz for example 2.

Figure 9 shows the two-sigma error estimate as a function of time for the two examples. The error ranges from 8 mV near the origin to 0.25 mV near the end of the record. The two-sigma error for the raw data is 20 mV.

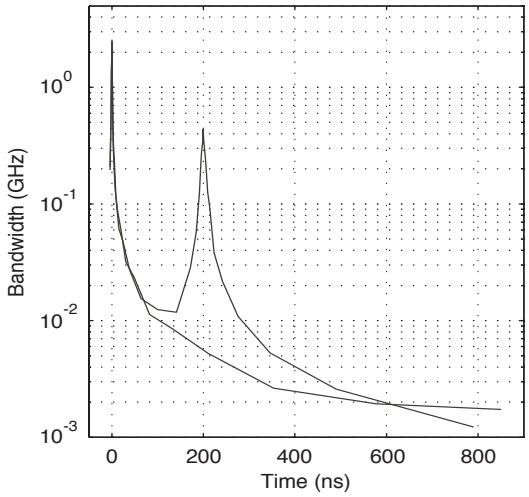


Fig. 8. The bandwidth as a function of time for examples 1 and 2.

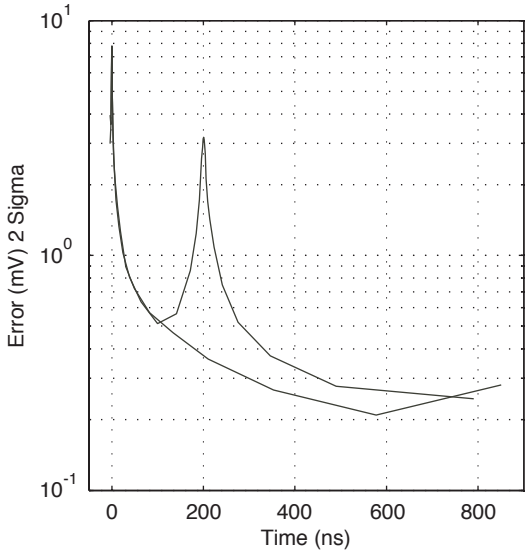


Fig. 9. The estimated 2-sigma error as a function of time for examples 1 and 2.

VIII. COMMENTS ON COMPUTING TIME

It was found, both experimentally and by analysis of the algorithm, that the time required to execute one iteration is approximately

$$time \cong C \times Nknots \times Nsamples \,, \tag{21}$$

where *Nknots* is the number of knots, *Nsamples* is the number of samples, and *C* is a constant that depends on the particular computing environment. The examples here were run on a 1.3 GHz Macintosh G4 computer running MATLAB, where the value of *C* was determined to be 2×10^{-6} s. Since the first iteration was done with 1000 knots, the time for the first iteration is $2\times10^{-6}\times1000\times40000$ s = 80 s. Thus, the first iteration took over one minute. The final number of knots for example 1 was 26, and for example 2 it was 42. Thus, the time to execute the last iteration for example 2 was about 3.4 s. It was also found that after each iteration the number of knots was no less than about 70% of the number on the previous iteration. The reason for this is not yet understood. This means that many iterations were performed with a much larger number of knots than the final value. Thus, a good initial guess at the knot sequence would yield a great improvement in computational efficiency. Good starting points for obtaining an initial guess are given in [12] and [13].

IX. CONCLUSIONS

An algorithm was presented that performs automatic smoothing of data from a digital oscilloscope to reduce the effects of noise. It is based on least-squares fitting to cubic spline functions. It requires that the data be over sampled and the noise be white. It requires no prior knowledge of the signal but that it has four derivatives. It automatically varies the bandwidth of the filter as a function of time and generates accurate estimates of the error.

REFERENCES

[1] Wiener, N., *Time Series*, M. I. T. Press, 1949.

[2] Kolmogorov, A., "Interpolation und extrapolation von stationären zufälligen folgen," *Bulletin de l'académie des sciences de U.R.S.S.*, Ser. Math., pp. 3–14, 1941.

[3] Kosulajeff, P., "Sur les problèmes d'interpolation et d'extraplation des suites stationnaires," *Comptes rendus de l'académie des sciences des U.R.S.S.*, vol. 30, pp. 13–17, 1941.

[4] Gauss, K., *Theory of the Combination of Observations Least Subject to Errors,* SIAM, 1995.

[5] Blair, J., "Error estimates derived from the data for least-squares spline fitting," *Instrumentation and Measurement Technology Conference Proceedings*, 2007 IEEE, pp. 1–6.

[6] Kay, S., *Fundamentals of Statistical Signal Processing – Estimation Theory*, Prentice Hall, 1993.

[7] Wahba, G., *Spline Model for Observational Data,* SIAM, 1990.

[8] de Boor, C., *A Practical Guide to Splines – Revised Edition*, Springer, 2001.

[9] *Spline Toolbox*, The Mathworks.

[10] Unser, M., A. Aldroubi, and M. Eden, "Polynomial spline signal approximations: filter design and asymptotic equivalence with Shannon's sampling theorem," *IEEE Trans. Info. Theory*, vol. 38, pp. 95–103, Feb. 1992.

[11] Unser, M., A. Aldroubi, and M. Eden, "B-spline signal processing: part I – theory,"*IEEE Trans. Signal Processing*, vol. 41, pp. 821–832, Feb. 1993.

[12] He, X., L. Shen and Z. Shen, "A data-adaptive knot selection scheme for fitting splines," *Signal Proc. Letters*, vol. 8, no. 5, pp. 137–139, May 2001.

[13] Ainsleigh, P. and C. Chui, "Simultaneous wavelet and spline smoothing of noisy data," *Acoustics, Speech and Signal Processing, 1993 IEEE International Conference on*, vol. 3, pp. 197–200, Apr. 1993.